AL/CF-TR-1995-0120

EFFICIENT COORDINATION OF AN ANTHROPOMORPHIC TELEMANIPULATION SYSTEM

Ming Z. Huang

CREW SYSTEMS DIRECTORATE
BIOACOUSTICS AND BIOCOMMUNICATIONS DIVISION
WRIGHT-PATTERSON AFB OH 45433-7901

DECEMBER 1993

INTERIM REPORT FOR THE PERIOD JUNE 1993 TO AUGUST 1993

AIR FORCE MATERIEL COMMAND
WRIGHT-PATTERSON AIR FORCE BASE, OHIO 45433-6573

ARMSTRONG LABORATORY

NOTICE

When US Government drawings, specifications, or other data are used for any purpose other than a definitely related Government procurement operation, the Government thereby incurs no responsibility nor any obligation whatsoever, and the fact that the Government may have formulated, furnished, or in any way supplied the said drawings, specifications, or other data, is not to be regarded by implication or otherwise, as in any manner, licensing the holder or any other person or corporation, or conveying any rights or permission to manufacture, use or sell any patented invention that may in any way be related thereto.

Please do not request copies of this report from the Armstrong Laboratory. Additional copies may be purchased from:

National Technical Information Service
5285 Port Royal Road
Springfield VA 22161

Federal Government agencies and their contractors registered with Defense Technical Information Center should direct requests for copies of this report to:

Defense Technical Information Center
Cameron Station
Alexandria VA 22314

TECHNICAL REVIEW AND APPROVAL

AL/CF-TR-1995-0120

This report has been reviewed by the Office of Public Affairs (PA) and is releasable to the National Technical Information Service (NTIS). At NTIS, it will be available to the general public, including foreign nations.

This technical report has been reviewed and is approved for publication.

FOR THE COMMANDER

THOMAS J. MOORE, Chief
Biodynamics and Biocommunications Division
Crew Systems Directorate
Armstrong Laboratory

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY *(Leave blank)*	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	DECEMBER 1993	INTERIM REPORT - JUNE 1993 TO AUGUST 1993

4. TITLE AND SUBTITLE
Efficient Coordination of an Anthropomorphic Telemanipulation System

5. FUNDING NUMBERS
PE - 65502F
PR - 7231
TA - 38
WU - 08

6. AUTHOR(S)
Ming Z. Huang

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Armstrong Laboratory, Crew Systems Directorate
Bioacoustics and Biocommunications Division
Human Systems Center
Air Force Materiel Command
Wright-Patterson AFB OH 45433-7901

8. PERFORMING ORGANIZATION REPORT NUMBER
AL/CF-TR-1995-0120

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution is unlimited.

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*

This report documents the development of coordination algorithms for control implementation of an anthropomorphic telemanipulation system presently at Wright-Patterson Air Force Base. The telemanipulation system, which is to be used as a research platform in facilitating studies on human sensory feedback, comprises a 7 degree-of-freedom (d.o.f.) force-reflecting exoskeleton master and a 6 d.o.f. articulated slave robot. The approach taken in the development emphasizes the practical issue of computation efficiency--a primary concern for satisfactory real-time operations. The algorithms presented here have been fully tested and implemented on the system. Implementation results indicate that at least a five-fold improvement on the control sampling rate has been achieved (from 11 Hz to 62 Hz on a 68020-based VME board). Other practical issues of implementation are also discussed.

14. SUBJECT TERMS
kinematics, Jacobian, bioengineering, telepresence, exoskeleton, robotics, man-machine systems, force-reflecting display

15. NUMBER OF PAGES
33

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	UNLIMITED

This page intentionally left blank.

DISCLAIMER NOTICE

THIS DOCUMENT IS BEST QUALITY AVAILABLE. THE COPY FURNISHED TO DTIC CONTAINED A SIGNIFICANT NUMBER OF PAGES WHICH DO NOT REPRODUCE LEGIBLY.

PREFACE

The support of this work as part of Summer Faculty Research Program sponsored by the Air Force Office of Scientific Research (AFOSR) is gratefully acknowledged. The author also wishes to express his appreciation for help received from Armstrong Laboratory, Bioacoustics and Biocommunications Branch (AL/CFBA) at Wright-Patterson Air Foce Base, Ohio. Special thanks go to members of the Human Sensory Feedback group: Capt. P. Whalen, Capt. D. Nelson, Lt. C. Hasser, Mr. M. Crabill and Mr. T. Mosher.

TABLE OF CONTENTS

LIST OF FIGURES

This page intentionally left blank.

INTRODUCTION

Recently, studies in telepresence, which involve man-in-the-loop control of sensory-rich, remotely operated robotic systems, have emerged as a new critical area of research and development. This is due to the increasing recognition and acceptance of telerobotic manipulation technology as being a viable solution for remote operations in unstructured and/or hazardous environments, such as space, undersea, or nuclear sites [1-3].

For years, man-machine interface research has been a main thrust of the efforts in the Air Force to improve performance and effectiveness of its crews. Under the Crew Systems Directorate, the Armstrong Laboratory at Wright-Patterson AFB conducts the Human Sensory Feedback (HSF) research program, which is charged with the mission to investigate telepresence and its related issues. Among the on-going research activities of the HSF program, main emphasis is currently concentrated on characterization of the role of human sensory feedback in the following three key aspects: coarse positioning (large scale motion associated with the human arm and wrist), fine manipulation (small scale motion associated with the human hand), and tactile feedback.

A unique telemanipulation test platform has been designed and built to support the research in the coarse manipulative human sensory feedback. The platform is unique in that it uses a custom-built, anthropomorphic exoskeleton capable of force reflection as the master control arm to command a kinematically dissimilar slave (a revolute-jointed industrial type) robot. The Force-REFLecting EXoskeleton (FREFLEX) master is a seven degree-of-freedom, cable-driven robot that was designed specifically to provide mobility and range of motion similar to that of a human arm (Figure 1). The use of such an anthropomorphic exoskeleton as the master controller enables the operator to generate motions and to react to forces encountered during manipulation in a natural way, a key functional requirement in human sensory feedback studies.

Force reflection on the FREFLEX is achieved by its controller generating appropriate antagonistic actions through cables (tendons) and pulleys, driven by brushless DC motors. While use of cables and pulleys makes it possible to drive distal links from actuators mounted at the base, significantly reducing overall weight of the arm, such an actuation scheme results in severe cross-coupling between motions of the joints and, consequently, there is coupling between joint torque commands. Given a certain

Figure 1. The Force REFLecting EXoskeleton (FREFLEX) Master Robot

force/moment to be reflected by the FREFLEX, it is necessary to identify the coupling relationship in order to compute actual torque commands at the actuators.

When performing telemanipulation, it is desirable that the master controller appears as being "weightless" to the operator. Besides the obvious benefit of reducing operator fatigue, gravity and inertia compensation increases the fidelity of manipulative interaction and hence the overall system performance. Update rate in the control system is another factor which affects the overall system performance. The system will feel sluggish or even become unstable if the cycle time is too large. In general, it is necessary to have an update rate of 20 to 200 Hz to ensure satisfactory real-time performance.

In this report, we present analyses and solutions to the problems of coordination concerning the control implementation of the telemanipulation system as described above. Specifically, kinematic models for both the FREFLEX and the slave (MERLIN^{TM} 6500 by American Robot Corp.) and their related kinematics solutions (both position and velocity) are developed. In addition, algorithms for the FREFLEX to compensate gravity loads and compute joint torques, including identification of the kinematic coupling relationship for the actuators and the joints, is also presented. It is noted that throughout the following development of algorithms we have taken special care to optimize computational efficiency. As a result, a five-fold improvement of the overall system update rate has been achieved with their implementation.

COORDINATION OF THE FREFLEX MASTER ROBOT

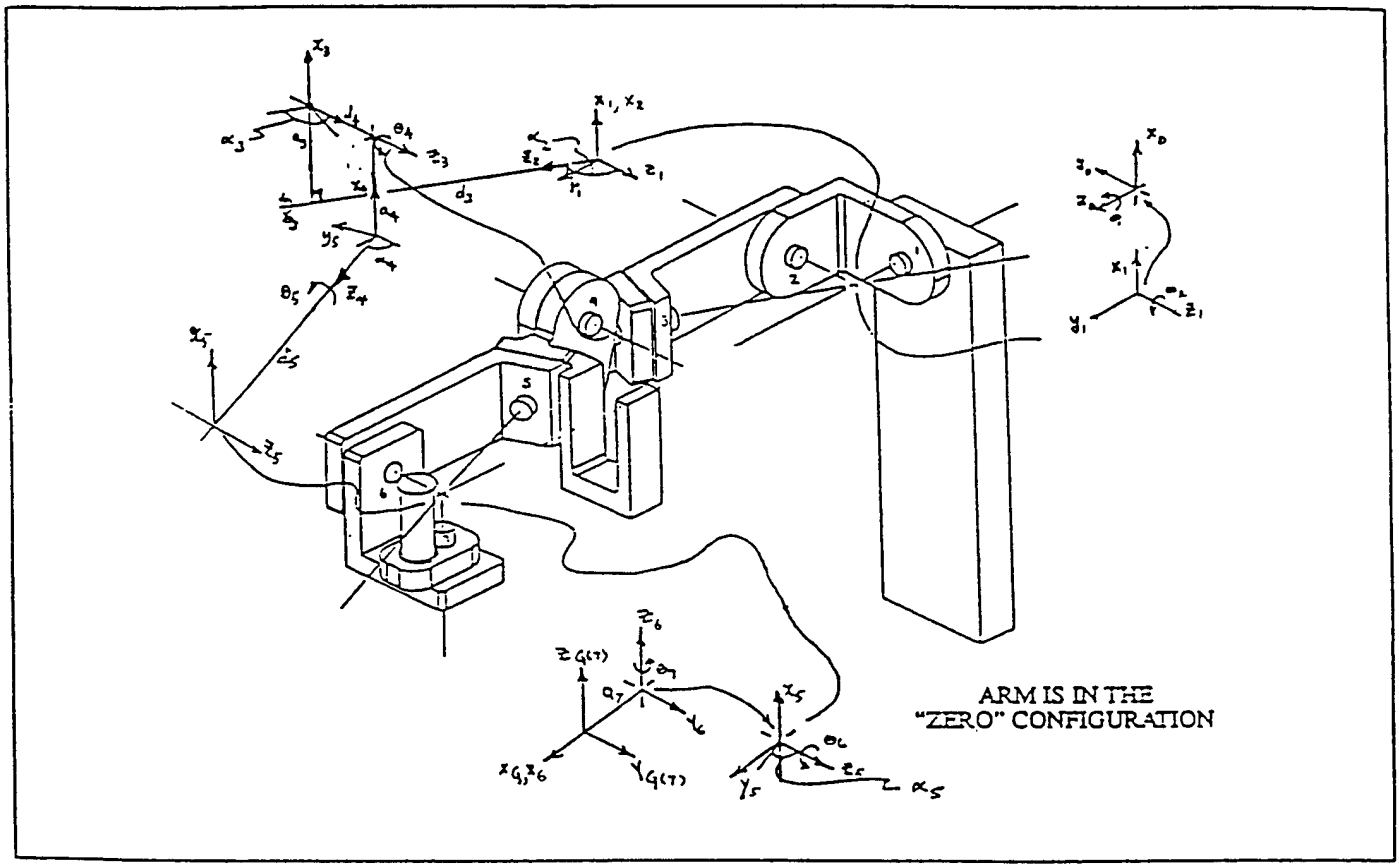
Coordination for the FREFLEX can be divided into the following stages of computation: forward position kinematics, Jacobian, gravity compensation, kinematic coupling, and joint torque decomposition. Note that for a single microprocessor system all of these computations must be completed before the next update to the controller can be made; in other words, the system sampling rate is dictated by the overall efficiency of these computations. Consequently, when formulating the solution to each stage, a sensible guideline would be to develop the algorithms in such a way that all items should be computed only once, and any computation which occurs in the later stages should take maximum advantage of what has been computed previously.

Forward Position Kinematics

Forward position kinematics refers to the following problem: given a set of measured joint positions, compute the position and orientation of the FREFLEX hand grip (or any conveniently chosen point of interest) in the Cartesian space. The resulting pose (combined position and orientation) is then used in the inverse kinematics solution for the slave robot (to be described later) to yield the corresponding joint commands for the slave to be driven to that same pose. Note that this transformation of position command in Cartesian space is necessary whenever the master and the slave robots are of different kinematic structures.

To establish a kinematic model for the FREFLEX, we adopt the so-called Denavit-Hartenberg (D-H) modeling convention with the frame assignment scheme similar to that adopted in [4] and [5]. Figure 2 shows a schematic of the FREFLEX with definitions of all the kinematic frames and a table of their corresponding D-H parameters. Based on the above D-H convention, a 4x4 homogeneous transformation, denoted as ${}^{i-1}T_i$, can be derived to represent the position and orientation of frame i relative to the frame (i-1); refer to [4] or [5]. It can be easily shown that the coordinate transformation representing position and orientation of the last frame (frame 7 for the FREFLEX) relative to the base frame (frame 0) can be obtained as:

$$ {}^{0}\mathbf{T}_7 = \prod_{i=1}^{7} {}^{i-1}\mathbf{T}_i = {}^{0}\mathbf{T}_1 \, {}^{1}\mathbf{T}_2 \cdots {}^{6}\mathbf{T}_7 \qquad (1) $$



	a_i	d_i	α_i	θ_i
1	0	0	90^o	θ_1
2	0	0	-120^o	θ_2
3	1.969"	14.640"	$+120^o$	θ_3
4	-1.969"	0.625"	-70^o	θ_4
5	0	11.770"	$+70^o$	θ_5
6	0	0	$+90^o$	θ_6
7	1.68"	0	0^o	θ_7

Figure 2. Schematic of the FREFLEX and Definitions of Kinematic Frames with Corresponding Table of D-H Parameters

or equivalently, in terms of ${}^{i-1}\mathbf{R}_i$ and ${}^{i-1}\mathbf{q}_i$, as:

$$
\begin{aligned}
{}^0\mathbf{R}_7 &= {}^0\mathbf{R}_1\,{}^1\mathbf{R}_2 \cdots {}^6\mathbf{R}_7 \\
{}^0\mathbf{q}_7 &= {}^0\mathbf{q}_1 + {}^0\mathbf{R}_1\,{}^1\mathbf{q}_2 + {}^0\mathbf{R}_2\,{}^2\mathbf{q}_3 + \cdots + {}^0\mathbf{R}_6\,{}^6\mathbf{q}_7
\end{aligned}
\qquad (2)
$$

where

$$
{}^{i-1}\mathbf{T}_i = \begin{bmatrix} c\theta_i & -s\theta_i c\alpha_i & s\theta_i s\alpha_i & a_i c\theta_i \\ s\theta_i & c\theta_i c\alpha_i & -c\theta_i s\alpha_i & a_i s\theta_i \\ 0 & s\alpha_i & c\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} {}^{i-1}\mathbf{R}_i & {}^{i-1}\mathbf{q}_i \\ 0\ 0\ 0 & 1 \end{bmatrix}
$$

It is remarked that, while both equations (1) and (2) are completely equivalent, Eq. (2) results in better computation efficiency by separating the computations into rotational and translational parts. By the author's own experience, the two-part form of Eq. (2) has also been found to be more amenable to manipulate symbolically, particularly in deriving the inverse kinematics solution. For example, in the FREFLEX, from its parameter table we have: ${}^0\mathbf{q}_1 = {}^1\mathbf{q}_2 = {}^5\mathbf{q}_6 = \mathbf{0}$. This immediately leads to a simplification on the translational part of Eq. (2) to: ${}^0\mathbf{q}_7 = {}^0\mathbf{R}_2\,{}^2\mathbf{q}_3 + {}^0\mathbf{R}_3\,{}^3\mathbf{q}_4 + {}^0\mathbf{R}_4\,{}^4\mathbf{q}_5 + {}^0\mathbf{R}_6\,{}^6\mathbf{q}_7$ which is clearly easier to manipulate, although the rotational part remains unchanged. The inverse kinematics solution for the slave robot (MERLIN), to be included later, was also reached based on the above procedure.

Note that if, instead of the last frame (frame 7), the position and orientation of frame k is to be computed, both of the above equations still hold with only a change in the upper index from 7 to k needed. It can be seen that an efficient way to compute the pose of any link frame is to do so sequentially by starting with k=1 (from the base) and then progressing outward. In fact, it is also desirable in practice to facilitate the forward kinematics computations so that the positions and orientations of all the link frames (not just the hand grip) are easily accessible should they be needed in subsequent computations. Based on Eq. (2), an outward iteration algorithm which computes the pose of each link frame starting with the base frame can be implemented using the following recursive relationships (i = 1 to k):

$$
\begin{aligned}
{}^0\mathbf{q}_i &= {}^0\mathbf{q}_{i-1} + {}^0\mathbf{R}_{i-1}\,{}^{i-1}\mathbf{q}_i \\
{}^0\mathbf{R}_i &= {}^0\mathbf{R}_{i-1}\,{}^{i-1}\mathbf{R}_i
\end{aligned}
\qquad (3)
$$

Knowing the position and orientation of each link frame (relative to the base), it is now straightforward to obtain the absolute position of any given point in any link. Let ${}^{i}\mathbf{p}$ be the position vector of a point in link frame i; then its corresponding position in the base frame, ${}^{0}\mathbf{p}$, is given by:

$$ {}^{0}\mathbf{p} = {}^{0}\mathbf{R}_{i}\,{}^{i}\mathbf{p} + {}^{0}\mathbf{q}_{i} \qquad (4) $$

In the FREFLEX implementation, a forward kinematics solution has been developed using the above recursive scheme. The solution for the pose of each link frame is obtained in the form of explicit analytic expressions for optimal efficiency. It is recognized that one may achieve reasonable efficiency by direct numerical computations using the same recursive scheme with a carefully coded algorithm. However, to guarantee optimal efficiency one must ensure all necessary terms are computed once and only once, which requires being able to identify and eliminate redundant, repetitious computations of those common terms which may be embedded in more than one expression. This can be accomplished only by going through the process of analytic derivation. It is recommended that this be done whenever efficiency is a critical factor in system performance.

The following describes the details of the derivation and the results of the forward kinematics solution for the FREFLEX. Referring to the D-H parameter table of FREFLEX (Figure 2), the corresponding link transformations between frame (i-1) and i are:

$$ {}^{0}\mathbf{R}_{1} = \begin{bmatrix} c\theta_1 & 0 & s\theta_1 \\ s\theta_1 & 0 & -c\theta_1 \\ 0 & 1 & 0 \end{bmatrix}; \quad {}^{1}\mathbf{R}_{2} = \begin{bmatrix} c\theta_2 & .5s\theta_2 & bs\theta_2 \\ s\theta_2 & -.5c\theta_2 & -bc\theta_2 \\ 0 & b & -.5 \end{bmatrix}; $$

$$ {}^{2}\mathbf{R}_{3} = \begin{bmatrix} c\theta_3 & .5s\theta_3 & -bs\theta_3 \\ s\theta_3 & -.5c\theta_3 & bc\theta_3 \\ 0 & -b & -.5 \end{bmatrix}; \quad {}^{3}\mathbf{R}_{4} = \begin{bmatrix} c\theta_4 & -hs\theta_4 & fs\theta_4 \\ s\theta_4 & hc\theta_4 & -fc\theta_3 \\ 0 & f & h \end{bmatrix}; $$

$$ {}^{4}\mathbf{R}_{5} = \begin{bmatrix} c\theta_5 & -hs\theta_5 & -fs\theta_5 \\ s\theta_5 & hc\theta_5 & fc\theta_5 \\ 0 & -f & h \end{bmatrix}; \quad {}^{5}\mathbf{R}_{6} = \begin{bmatrix} c\theta_6 & 0 & s\theta_6 \\ s\theta_6 & 0 & -c\theta_6 \\ 0 & 1 & 0 \end{bmatrix}; \quad {}^{6}\mathbf{R}_{7} = \begin{bmatrix} c\theta_7 & -s\theta_7 & 0 \\ s\theta_7 & c\theta_7 & 0 \\ 0 & 0 & 1 \end{bmatrix} $$

and

$$^0\mathbf{q}_1 = {}^1\mathbf{q}_2 = {}^5\mathbf{q}_6 = \mathbf{0}$$

$$^2\mathbf{q}_3 = \begin{bmatrix} 1.969\,c\theta_3 \\ 1.969\,s\theta_3 \\ 14.640 \end{bmatrix}; \quad ^3\mathbf{q}_4 = \begin{bmatrix} -1.969\,c\theta_4 \\ -1.969\,s\theta_4 \\ .625 \end{bmatrix}; \quad ^3\mathbf{q}_4 = \begin{bmatrix} 0 \\ 0 \\ 11.770 \end{bmatrix}; \quad ^6\mathbf{q}_7 = \begin{bmatrix} 1.68\,c\theta_7 \\ 1.68\,s\theta_7 \\ x \end{bmatrix}$$

where

$$b = \sin(-120°); \quad h = \cos(-70°); \quad f = \sin(70°)$$

and

$$x = \begin{cases} 0, & \text{if frame 7 is at the base of the handgrip.} \\ \frac{1}{2}L_{grip}, & \text{if frame 7 is at the center of the handgrip} \end{cases}$$

Following the algorithm given by Eq. (3), with i =1 to 7, the transformations in the form directly usable in Eq. (4) are derived symbolically below.

i =1:

$$^0\mathbf{q}_1 = \mathbf{0}; \quad ^0\mathbf{R}_1 = \begin{bmatrix} c_1 & 0 & s_1 \\ s_1 & 0 & -c_1 \\ 0 & 1 & 0 \end{bmatrix}$$

i =2:

$$^0\mathbf{q}_2 = \mathbf{0};$$

$$^0\mathbf{R}_2 = \begin{bmatrix} c_1 & 0 & s_1 \\ s_1 & 0 & -c_1 \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} c_2 & .5s_2 & bs_2 \\ s_2 & -.5c_2 & -bc_2 \\ 0 & b & -.5 \end{bmatrix}$$

$$= \begin{bmatrix} c_1c_2 & .5c_1s_2 + bs_1 & bc_1s_2 - .5s_1 \\ s_1c_2 & .5s_1s_2 - bc_1 & bs_1s_2 + .5c_1 \\ s_2 & -.5c_2 & -bc_2 \end{bmatrix} \equiv \begin{bmatrix} b_{11} & b_{12} & b_{13} \\ b_{21} & b_{22} & b_{23} \\ b_{31} & b_{32} & b_{33} \end{bmatrix}$$

i =3:

$$ {}^0\mathbf{q}_3 = {}^0\mathbf{R}_2\,{}^2\mathbf{q}_3 = \begin{bmatrix} 1.969e_{11} + 14.64b_{13} \\ 1.969e_{21} + 14.64b_{23} \\ 1.969e_{31} + 14.64b_{33} \end{bmatrix} \equiv \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} $$

$$ {}^0\mathbf{R}_3 = \begin{bmatrix} b_{11} & b_{12} & b_{13} \\ b_{21} & b_{22} & b_{23} \\ b_{31} & b_{32} & b_{33} \end{bmatrix} \begin{bmatrix} c_3 & .5s_3 & -bs_3 \\ s_3 & -.5c_3 & bc_3 \\ 0 & -b & -.5 \end{bmatrix} $$

$$ = \begin{bmatrix} e_{11} & .5e_{12} - bb_{13} & be_{12} - .5b_{13} \\ e_{21} & .5e_{22} - bb_{23} & be_{22} - .5b_{23} \\ e_{31} & .5e_{32} - bb_{33} & be_{32} - .5b_{33} \end{bmatrix} \equiv \begin{bmatrix} u_{11} & u_{12} & u_{13} \\ u_{21} & u_{22} & u_{23} \\ u_{31} & u_{32} & u_{33} \end{bmatrix} $$

where $e_{11} = b_{11}c_3 + b_{12}s_3$ $\quad e_{21} = b_{21}c_3 + b_{22}s_3$ $\quad e_{31} = b_{31}c_3 + b_{32}s_3$

$e_{12} = b_{11}s_3 - b_{12}c_3$ $\quad e_{22} = b_{21}s_3 - b_{22}c_3$ $\quad e_{32} = b_{31}s_3 - b_{32}c_3$

i = 4:

$$ {}^0\mathbf{q}_4 = {}^0\mathbf{q}_3 + {}^0\mathbf{R}_3\,{}^3\mathbf{q}_4 = \begin{bmatrix} u_1 - 1.969r_{11} + .625u_{13} \\ u_2 - 1.969r_{21} + .625u_{23} \\ u_3 - 1.969r_{31} + .625u_{33} \end{bmatrix} \equiv \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix} $$

$$ {}^0\mathbf{R}_4 = {}^0\mathbf{R}_3\,{}^3\mathbf{R}_4 = \begin{bmatrix} u_{11} & u_{12} & u_{13} \\ u_{21} & u_{22} & u_{23} \\ u_{31} & u_{32} & u_{33} \end{bmatrix} \begin{bmatrix} c_4 & -hs_4 & fs_4 \\ s_4 & hc_4 & -fc_3 \\ 0 & f & h \end{bmatrix} $$

$$ = \begin{bmatrix} r_{11} & -hr_{12} + fu_{13} & fr_{12} + hu_{13} \\ r_{21} & -hr_{22} + fu_{23} & fr_{22} + hu_{23} \\ r_{31} & -hr_{31} + fu_{33} & fr_{31} + hu_{33} \end{bmatrix} \equiv \begin{bmatrix} v_{11} & v_{12} & v_{13} \\ v_{21} & v_{22} & v_{23} \\ v_{31} & v_{32} & v_{33} \end{bmatrix} $$

where $r_{11} = u_{11}c_4 + u_{12}s_4$ $\quad r_{21} = u_{21}c_4 + u_{22}s_4$ $\quad r_{31} = u_{31}c_4 + u_{32}s_4$

$r_{12} = u_{11}s_4 - u_{12}c_4$ $\quad r_{22} = u_{21}s_4 - b_{22}c_4$ $\quad r_{32} = u_{31}s_4 - u_{32}c_4$

i = 5:

$$ {}^0\mathbf{q}_5 = {}^0\mathbf{q}_4 + {}^0\mathbf{R}_4\,{}^4\mathbf{q}_5 = \begin{bmatrix} v_1 + 11.77v_{13} \\ v_2 + 11.77v_{23} \\ v_3 + 11.77v_{33} \end{bmatrix} \equiv \begin{bmatrix} w_1 \\ w_2 \\ w_3 \end{bmatrix} $$

$$ {}^0\mathbf{R}_5 = {}^0\mathbf{R}_4\,{}^4\mathbf{R}_5 = \begin{bmatrix} v_{11} & v_{12} & v_{13} \\ v_{21} & v_{22} & v_{23} \\ v_{31} & v_{32} & v_{33} \end{bmatrix} \begin{bmatrix} c_5 & -hs_5 & -fs_5 \\ s_5 & hc_5 & fc_5 \\ 0 & -f & h \end{bmatrix} $$

$$ = \begin{bmatrix} v_{11}c_5 + v_{12}s_5 & ht_{11} - fv_{13} & ft_{11} + hv_{13} \\ v_{21}c_5 + v_{22}s_5 & ht_{21} - fv_{23} & ft_{21} + hv_{23} \\ v_{31}c_5 + v_{32}s_5 & ht_{31} - fv_{33} & ft_{31} + hv_{13} \end{bmatrix} \equiv \begin{bmatrix} w_{11} & w_{12} & w_{13} \\ w_{21} & w_{22} & w_{23} \\ w_{31} & w_{32} & w_{33} \end{bmatrix} $$

where $t_{11} = v_{12}c_5 - v_{11}s_5 \qquad t_{21} = v_{22}c_5 - v_{21}s_5 \qquad t_{31} = v_{32}c_5 - v_{31}s_5$

i = 6:

$$ {}^0\mathbf{q}_6 = {}^0\mathbf{q}_5 = \begin{bmatrix} w_1 \\ w_2 \\ w_3 \end{bmatrix} $$

$$ {}^0\mathbf{R}_6 = {}^0\mathbf{R}_5\,{}^5\mathbf{R}_6 = \begin{bmatrix} w_{11} & w_{12} & w_{13} \\ w_{21} & w_{22} & w_{23} \\ w_{31} & w_{32} & w_{33} \end{bmatrix} \begin{bmatrix} c_6 & 0 & s_6 \\ s_6 & 0 & -c_6 \\ 0 & 1 & 0 \end{bmatrix} $$

$$ = \begin{bmatrix} w_{11}c_6 + w_{12}s_6 & w_{13} & w_{11}s_6 - w_{12}c_6 \\ w_{21}c_6 + w_{22}s_6 & w_{23} & w_{21}s_6 - w_{22}c_6 \\ w_{31}c_6 + w_{32}s_6 & w_{33} & w_{31}s_6 - w_{32}c_6 \end{bmatrix} \equiv \begin{bmatrix} y_{11} & y_{12} & y_{13} \\ y_{21} & y_{22} & y_{23} \\ y_{31} & y_{32} & y_{33} \end{bmatrix} $$

i = 7:

$$ {}^0\mathbf{q}_7 = {}^0\mathbf{q}_6 + {}^0\mathbf{R}_6\,{}^6\mathbf{q}_7 = \begin{bmatrix} w_1 + 1.68z_{11} + xy_{13} \\ w_2 + 1.68z_{21} + xy_{23} \\ w_3 + 1.68z_{31} + xy_{33} \end{bmatrix} $$

$$ {}^0\mathbf{R}_7 = {}^0\mathbf{R}_6\,{}^6\mathbf{R}_7 = \begin{bmatrix} y_{11} & y_{12} & y_{13} \\ y_{21} & y_{22} & y_{23} \\ y_{31} & y_{32} & y_{33} \end{bmatrix} \begin{bmatrix} c_7 & -s_7 & 0 \\ s_7 & c_7 & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} z_{11} & -y_{11}s_7 + y_{12}c_7 & y_{13} \\ z_{21} & -y_{21}s_7 + y_{22}c_7 & y_{23} \\ z_{31} & -y_{31}s_7 + y_{32}c_7 & y_{33} \end{bmatrix} $$

where $z_{11} = y_{11}c_7 + y_{12}s_7 \quad z_{21} = y_{21}c_7 + y_{22}s_7 \quad z_{31} = y_{31}c_7 + y_{32}s_7$

It is noted again that all terms which appear more than once have been identified, and upon computations, their values are stored as new parameters to avoid redundant

operations. As alluded to earlier, this is only possible when the forward kinematics transformations are derived analytically. We have evaluated the computational efficiency of our algorithm in terms of the required operation counts for addition/subtraction (A), multiplication/division (M), and trigonometric function calls (F). For the FREFLEX, the forward kinematics algorithm calls for a total of (64A+133M+14F), as compared to the (162A+216M+14F) needed if implemented with direct numerical computation using Eq. (3).

Jacobian Formulation

As is well known in robotics, the Jacobian is a transformation matrix which relates differential motions (linear and angular) of the robot end-effector in the Cartesian space to the corresponding differential displacements at the joints. It is also known that in statics consideration the same transformation can be used to relate the external force and moment applying at the end-effector to the torques (or forces, if prismatic) at the joints. Mathematically, the above statements can be expressed using the following equations:

$$\mathbf{J}\,\dot{\theta} = \begin{bmatrix} \omega \\ \mathbf{v} \end{bmatrix} \tag{5}$$

$$\mathbf{J}^{T} \begin{bmatrix} \mathbf{M} \\ \mathbf{F} \end{bmatrix} = \tau \tag{6}$$

where

J: the manipulator Jacobian

w, and **v**: the end-effector angular and linear velocities, respectively

M and **F**: the resultant moment and force by the end-effector, respectively

$\dot{\theta}$: nx1 vector of joint rates (n = manipulator's degree of freedom)

τ: nx1 vector of joint torques/forces

and superscript T indicates the matrix transpose operation.

Although the above relationships may have been seen in many texts, it would be helpful to clarify a few points when considering their applications. First and foremost, care must be taken to ensure that all vector quantities involved be formulated with respect to the same coordinate frame of reference. In addition, point-specific vectors, such as linear velocity **v** and moment **M**, must be given such that they are all relative to the same point of reference which was used to formulate the Jacobian. In other words, the Jacobian may

take on various different forms, depending upon the reference frame and the reference point chosen (to describe **v** or **M**) during its formulation.

To emphasize the distinction between the various available forms, we shall denote the Jacobian with ${}^{m}\mathbf{J}_{k}$, in which the leading superscript, m, specifies the frame and the trailing subscript, k, gives the reference point. In general, ${}^{m}\mathbf{J}_{k}$ for a robot of n degrees of freedom can be formulated using the following:

$$ {}^{m}\mathbf{J}_{k} = \begin{bmatrix} \mathbf{u}_i \\ \rho_i \times \mathbf{u}_i \end{bmatrix}; \; i = 1,2,\ldots,n \qquad (7) $$

in which $\mathbf{u}_i$ is the unit directional vector of joint axis i, and ρ_i is the position vector of axis i with respect to the reference point k; of course, both vectors are expressed in frame m. Note that the above form of column vector in Eq. (7) only applies to robots with all revolute joints. In the case for robots with prismatic joints, say joint j, then the j-th column should be replaced with $\left[(0,0,0); \mathbf{u}_j^{T}\right]^{T}$.

For the FREFLEX, we have chosen to use ${}^{0}\mathbf{J}_{0}$, which is to formulate the Jacobian in the base frame (frame 0) with its origin as the reference point. Again, the main consideration here is to minimize computation cost. The reason to use ${}^{0}\mathbf{J}_{0}$ is two-fold: first, the origin of frame 0 is the point of concurrency of axes 1, 2, and 3; and second, all the terms required to formulate the Jacobian are readily available from the forward kinematics procedure with no further manipulation necessary. As a result, we arrived at the following 6x7 matrix as the Jacobian for the FREFLEX:

$$ {}^{0}\mathbf{J}_{0} = \begin{bmatrix} \hat{\mathbf{z}}_0 & \hat{\mathbf{z}}_1 & \hat{\mathbf{z}}_2 & \hat{\mathbf{z}}_3 & \hat{\mathbf{z}}_4 & \hat{\mathbf{z}}_5 & \hat{\mathbf{z}}_6 \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & {}^{0}\mathbf{q}_3 \times \hat{\mathbf{z}}_3 & {}^{0}\mathbf{q}_4 \times \hat{\mathbf{z}}_4 & {}^{0}\mathbf{q}_5 \times \hat{\mathbf{z}}_5 & {}^{0}\mathbf{q}_6 \times \hat{\mathbf{z}}_6 \end{bmatrix} \qquad (8) $$

where, referring to Figure 2, $\hat{\mathbf{z}}_{i-1} = \mathbf{u}_i$ (i= 1 to 7) which is the third column of ${}^{0}\mathbf{R}_{i}$, and ${}^{0}\mathbf{q}_{i}$ is, as defined before, the position of the origin of frame i on axis (i+1); all of which can be obtained directly from the results of the forward kinematics computations. As can be seen from Eq. (8), only four additional vector cross product operations (24M+12A) are required to obtain the Jacobian.

The Jacobian formulated here will be used in Eq. (6) to compute the necessary joint torques in order to "reflect" a certain external load (force and moment). Typically, the point

at which the load is to be reflected will not be the same as the reference point used by the Jacobian. Therefore, before Eq. (6) can be applied it is necessary to transform the generalized load vector (both force and moment) to the same reference point, which is at the base frame origin. For discussion sake, let the load reflection point be at the wrist center of the FREFLEX. Typically this will be the case since the external load will usually be measured using a force/torque sensor mounted at the wrist of the slave robot. Then the following transformations are required:

$$^{0}\mathbf{F}_{0} = {}^{0}\mathbf{R}_{s}\,{}^{s}\mathbf{F}_{w} \tag{9}$$

$$^{0}\mathbf{M}_{0} = {}^{0}\mathbf{R}_{s}\,{}^{s}\mathbf{M}_{w} + {}^{0}\mathbf{p}_{w} \times {}^{0}\mathbf{F}_{0} \tag{10}$$

where ${}^{s}\mathbf{F}_{w}$ and ${}^{s}\mathbf{M}_{w}$ denote the measured force and moment in the sensor frame (s), ${}^{0}\mathbf{R}_{s}$ is the coordinate transformation matrix from the sensor to the base frames, and ${}^{0}\mathbf{p}_{w}$ is the position of the load reflection point, namely wrist center, relative to base frame. One can easily generalize the above relationships to cases where the point of reflection is different, simply by using the corresponding position vector of the new load reflection point and, if necessary, the appropriate coordinate transformation matrix.

Gravity Compensation

To facilitate better utility and reduce operator fatigue, the FREFLEX needs to support its own weight as it is being moved about providing position commands to control the slave manipulator. In general, the capability of gravity compensation is necessary for any master robot used in telerobotic systems. During each sampling period, this requires computing first the gravity loads at all joints and subsequently the torques needed from the motors to statically counteract those gravity loads. In this section we only address the former, which is the computation of the gravity loads (torques) at the exoskeleton joints. The discussion of the latter, namely the decomposition of these computed joint torques into the actual motor torques, is deferred to the following sections.

Consider the schematic of link connection at a revolute joint in a serial chain as illustrated in Figure 3. The following observations can be made: As a result of the serial chain configuration, the gravity load seen by that joint will be due to the sum of moments generated by weights of all the links outboard from it (farther from the base). And the load will vary as relative positions of links change from one configuration to another; in other words, it is position dependent.

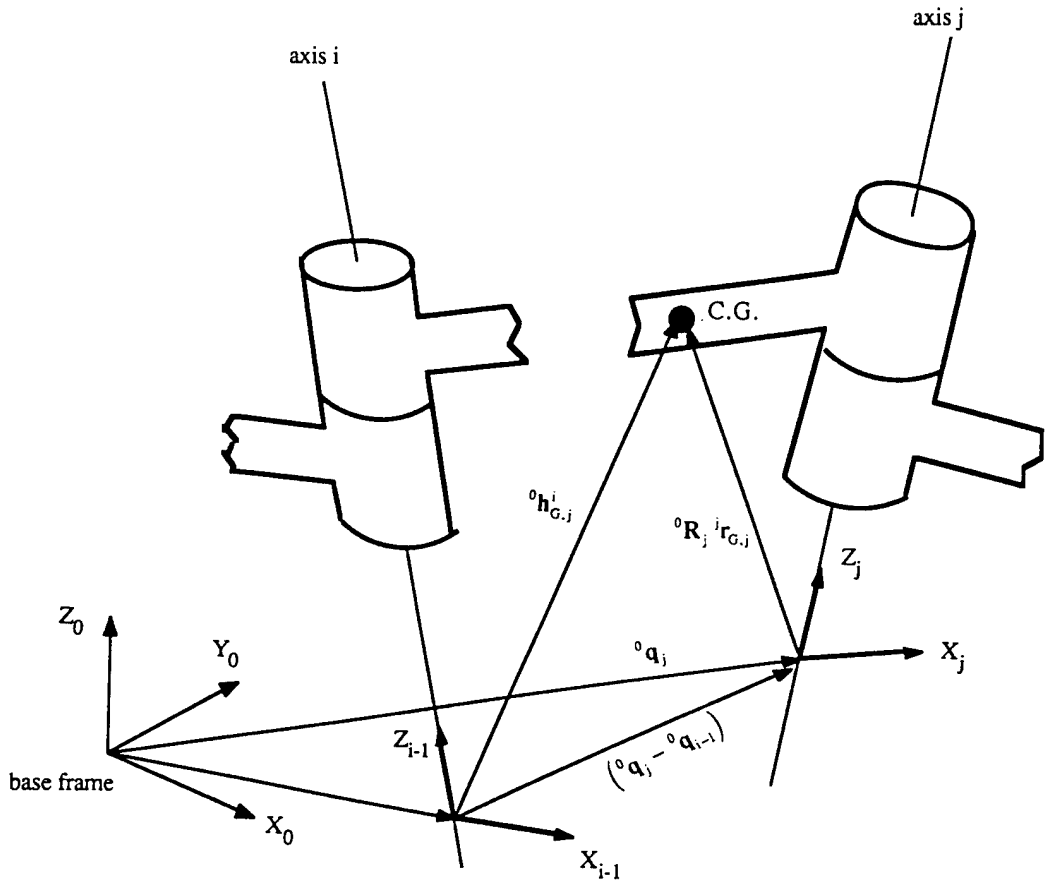


Figure 3. Kinematic Illustration of CG Position of j-th Link Relative to Axis I (Frame (i-1))

Let $\mathbf{h}^i_{G,j}$ represent the position vector of the center of gravity (CG) of link j relative to joint axis i (Figure 3) and the link mass be given as m_j. Then, with the origin of frame (i-1) as the point of reference, we can write the moment at joint i due to the weight of link j as:

$$\mathbf{n}_{i,j} = \mathbf{h}^i_{G,j} \times m_j \mathbf{g} \tag{11}$$

Thus, the total moment due to all outboard links ($j \geq i$) is:

$$\mathbf{n}_i = \sum_{j=i}^{7} \mathbf{n}_{i,j} = \sum_{j=i}^{7} \left(\mathbf{h}^i_{G,j} \times m_j \mathbf{g} \right) \tag{12}$$

And the corresponding joint torque (t_i) is just the component of total moment along the joint axis:

$$\tau_i = \mathbf{n}_i \cdot \hat{\mathbf{z}}_{i-1} \tag{13}$$

It can be easily seen that computing $\mathbf{h}^i_{G,j}$ is the key step in the gravity torque computation procedure. The way in which they are computed directly affects the efficiency of this algorithm. A logical consideration here is to take advantage of results available from the previous forward kinematics procedure. This suggests that the computations should be made in the base frame. We compute $\mathbf{h}^i_{G,j}$ using the following relationship, which can be easily arrived from Figure 3:

$${}^0\mathbf{h}^i_{G,j} = \left({}^0\mathbf{q}_j - {}^0\mathbf{q}_{i-1} \right) + {}^0\mathbf{R}_j \, {}^j\mathbf{r}_{G,j} ; \qquad j = i, \ldots, 7 \tag{14}$$

where ${}^j\mathbf{r}_{G,j}$ is the given position of CG for link j with respect to its local frame j, and ${}^0\mathbf{q}_j$ and ${}^0\mathbf{R}_j$ are, as defined before, position and orientation matrices of frame j, both of which are readily available from previous results.

To pursue the issue of efficiency a little further, computing in the base frame coordinate has associated with it another added benefit. For the cross product calculation in Eq. (11) it is only necessary to compute two out of the three components. This is because, when expressed in the base frame, the gravity vector will typically have a nonzero component along only one coordinate axis. For example, in terms of the FREFLEX base frame, $\mathbf{g} = [-g, 0, 0]^T$, and thus the cross product which results has only nonzero y and z components. As a result, Eqs. (12) and (13) for the FREFLEX can be evaluated using the

simplified component forms (note that, in what follows, the superscript 0 of the **h** terms given by Eq. (14) above has been dropped for convenience):

$$\mathbf{n}_i = \begin{cases} n_{i,x} = 0 \\ n_{i,y} = -g\sum_{j=i}^{7}\left(h^i_{j,z} m_j\right) \\ n_{i,z} = g\sum_{j=i}^{7}\left(h^i_{j,y} m_j\right) \end{cases} \tag{15}$$

and

$$\tau_i = n_{i,y} u_{i,y} + n_{i,z} u_{i,z} \tag{16}$$

Implementation of the above equations, Eqs. (14) to (16), has resulted in a highly efficient gravity torque computation algorithm for the FREFLEX which requires only (126M+133A+0F) in its calculations. In the following, we summarize the computation procedure of the gravity torque algorithm.

Procedure for Computing Gravity Torques

Step 1: Transform position of j-th link CG with respect to the i-th frame ($j \geq i$) in the base frame 0:

$$^0\mathbf{r}_j = {}^0\mathbf{R}_j\, {}^j\mathbf{r}_{G,j}; \qquad j = 1,2,\ldots,7$$

Step 2: Compute position of j-th link CG with respect to the i-th frame ($j \geq i$, where i = 1 to 7; and j = i to 7):

$$^0\mathbf{h}^i_{G,j} = \left({}^0\mathbf{q}_j - {}^0\mathbf{q}_{i-1}\right) + {}^0\mathbf{r}_j$$

e.g., let i=1, j=7:

$$^0\mathbf{h}^1_{G,1} = \left({}^0\mathbf{q}_1 - {}^0\mathbf{q}_0\right) + {}^0\mathbf{r}_1$$
$$^0\mathbf{h}^1_{G,2} = \left({}^0\mathbf{q}_2 - {}^0\mathbf{q}_0\right) + {}^0\mathbf{r}_2$$
$$\vdots$$
$$^0\mathbf{h}^1_{G,7} = \left({}^0\mathbf{q}_7 - {}^0\mathbf{q}_0\right) + {}^0\mathbf{r}_7$$

Step 3: Compute moment about the reference point on axis i; i.e., the origin of frame (i-1), due to the weights of all j-th links:

$$n_{i,y} = -g\sum_{j=i}^{7}\left(h^{i}_{j,z} m_j\right)$$

$$n_{i,z} = -g\sum_{j=i}^{7}\left(h^{i}_{j,y} m_j\right)$$

Step 4: The gravity torque for joint i is the component of the moment in the direction of its axis:

$$\tau_i = n_{i,y} u_{i,y} + n_{i,z} u_{i,z}$$

Based on the above computed results, ideally one should be able to completely compensate the weight of all the links. This of course assumes a-priori that the data on the inertia properties (link masses and CG locations) of the system are correct, which unfortunately is typically not the case. A practical method to deal with this problem is to experimentally "scale" the values of computed gravity torques. Such constant scaling factors can be found one joint at a time by adjusting from the last one inward until the link(s) supported by that joint in effect begin to "float".

Kinematic Coupling Relationship

As mentioned in the introduction, the cable and pulley actuation scheme gives rise to a complication of cross-coupling between motions at the joints on the FREFLEX and its actuators at the base. Figure 4 illustrates the cable system arrangement used by the FREFLEX [6]. In the present scheme, cables to drive a distal joint are routed via pulleys (idlers) through all inboard joints which are more proximal to the base. As a consequence, when the motor for the distal joint (the joint at which the cable terminates) is actuated, torque is transmitted not only to that distal joint, but to each of the inboard joints of the arm, which in turn causes motions at all those proximal joints. This is in contrast to the conventional actuation scheme in which the motion of a joint is dependent only on a single actuator driving that joint.

To control the FREFLEX, it is necessary to identify the above coupling relationship in order to "decouple" desired joint motions or torques into appropriate actuator commands. It should be noted that the coupling as described here is kinematic in nature in that it arises solely from kinematic constraints due to cable routing, and should not to be confused with the inertia coupling as seen in dynamics. Let $\Delta\theta_{motor}$ and $\Delta\theta_{joint}$ represent, respectively, the 7x1 vectors of displacements at the motors and their corresponding displacements at the

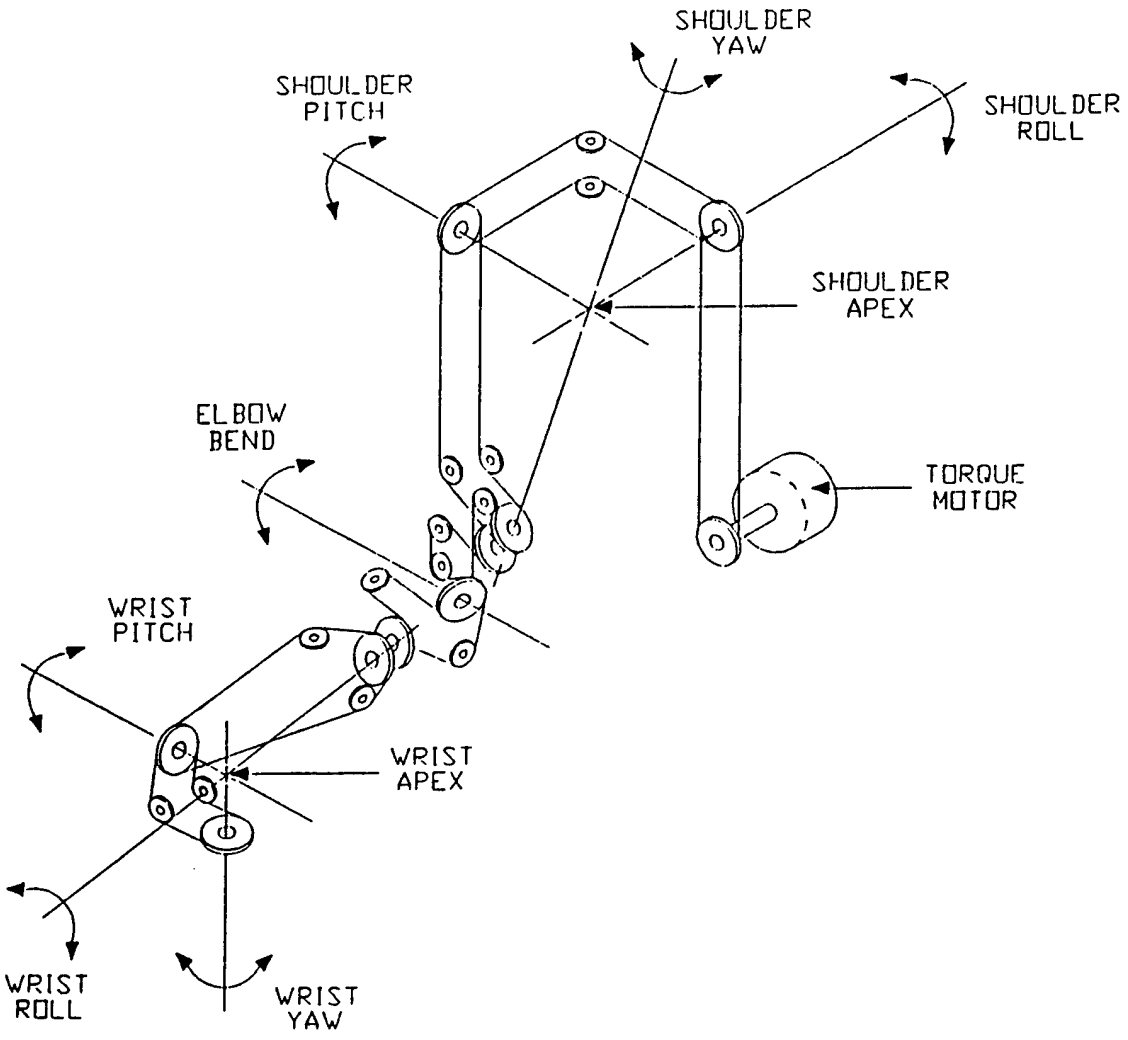


Figure 4. Cable Drive System of the FREFLEX

FREFLEX joints. Then they can be related by a (constant global) coupling matrix of 7x7, **A**, as follows:

$$\Delta\theta_{joint} = \mathbf{A} \cdot \Delta\theta_{motor} \tag{17}$$

The above coupling matrix is constant throughout the FREFLEX workspace, since it is only a function of geometric attributes such as relative locations of the joints and pulleys, numbers of pulleys used, and radii of pulleys [7]. Ideally, such a matrix can be expected to be lower triangular.

In theory one can attempt a direct approach to derive the coupling matrix based on the aforementioned geometric attributes. However, practical considerations--such as errors induced by uncertainties as well as inaccuracies associated with geometric data, suggested that in fact it would be more reliable to identify the coupling relationship indirectly, using actual data of joint and motor displacements measured experimentally.

Referring to Eq. (17), it can be easily seen that if sufficient measurements of $\Delta\theta_{motor}$ and $\Delta\theta_{joint}$ are available (in this case the minimum number is 7), then identification of the coupling matrix **A** amounts to solving a system of linear equations with its elements as the independent variables. Furthermore, one can minimize the effect of measurement noises by using more measurement data than the minimum and computing the generalized inverse solution of **A**. In fact, such a solution, to be given below, represents physically the optimum or best-fit set of elements for matrix **A** in a least-squares sense.

Let the number of measurements be, say, n ($n \geq 7$). By using the n measured joint displacement vectors as columns, a 7xn joint measurement matrix, defined as **C**, can be formed. Similarly, let **B** represent the corresponding 7xn actuator measurement matrix comprising the correspondingly ordered actuator displacement vectors as columns. Then we can obtain the least-squares solution of **A** by:

$$\mathbf{A} = \mathbf{C}\ \mathbf{B}^{T}\left(\mathbf{B}\mathbf{B}^{T}\right)^{-1} \tag{18}$$

To acquire the necessary joint and actuator displacement data, the following process was carried out on the FREFLEX. The FREFLEX was moved to take on different configurations within its workspace. In the meantime, actual position readings from potentiometers at the joints and resolvers at the actuators were recorded at a constant sampling interval. Having acquired the joint and motor position readings, the difference

between each consecutive set of data was computed to give the corresponding displacement vectors which can then be used in Eq. (18) to solve for the coupling matrix. It should be noted that during these data collection movements, it is important that care should be taken not to run any joint up to its mechanical limits in order to avoid artificial bias of joint data.

The following gives the result of the coupling matrix we have identified for the FREFLEX from a total of 16 position measurements, namely, 15 sets of displacement vectors:

$$\mathbf{A} = \begin{bmatrix} -.9617 & .0548 & -.0169 & .0194 & .0065 & -.0024 & -.0255 \\ .9632 & .9711 & .0039 & .0001 & .0045 & -.0029 & -.0046 \\ -.0246 & -.4294 & .9672 & .0257 & -.0231 & .0177 & -.0305 \\ .0292 & .1976 & .7246 & -.7250 & -.0278 & .0310 & -.0408 \\ .0075 & .4336 & .5890 & .3732 & -1.1949 & -.0052 & -.0032 \\ -.0052 & .3139 & -.2185 & .2192 & 1.2706 & -1.5005 & -.0062 \\ .0539 & .4949 & -.3120 & .3114 & .0092 & 1.4494 & -1.4256 \end{bmatrix}$$

As can be seen from the result obtained here, the coupling matrix is in general agreement with that expected from theory, that is, elements in its upper half of the matrix above the diagonal, while not ideally all zeros, all have relatively small magnitudes compared to those in the lower half. Furthermore, it can be seen that strong cross coupling exists between joints 5, 6, and 7 (as can be evidenced from unusually large values of (6,5) and (7,6) elements), a characteristic which has been regularly observed from actual behavior of the FREFLEX.

Joint Torque Decomposition

The same kinematic coupling matrix developed in the previous section can also be used to decouple the torques required at the FREFLEX joints into those needed to be applied at the motors. By principle of virtual work, a corresponding static coupling relationship can be reached in a straightforward way. This is illustrated as follows. Based on the input-output relation of (virtual) work, we must have

$$\tau_{motor} \cdot \Delta\theta_{motor} = \tau_{joint} \cdot \Delta\theta_{joint} \tag{19}$$

And through the use of Eq. (17), one can easily arrive at:

$$\tau_{motor} = \mathbf{A}^T \tau_{joint} \tag{20}$$

With Eq. (20), it is now possible to compute the torque commands which are necessary to be applied at the motors so as to (1) counteract the gravity loads due to their own weight, and (2) generate the external load (force and/or moment) to be reflected to the operator. Designating such motor torques in vector form as $\tau_{motor,total}$, we can combine Eqs. (6), (16), and (20) to yield:

$$\tau_{motor,total} = \mathbf{A}^T \left(-\tau_{gravity} + \mathbf{J}^T \begin{bmatrix} \mathbf{M}_{ext} \\ \mathbf{F}_{ext} \end{bmatrix} \right)$$

For clarity, the subscript "gravity" has been added to the first term in the parentheses to delineate the joint torque vector due only to gravity, which is available from Eq. (16) directly. Notice that this result is then negated to give the joint torques needed to equilibrate the gravity loads before its decomposition into corresponding motor commands. Similarly, the subscript "ext" in the second term represents resultant load of force and moment (as given by Eqs. (9) and (10)) to be generated by the motors, in addition to the gravity loads.

POSITION COORDINATION FOR THE SLAVE MANIPULATOR

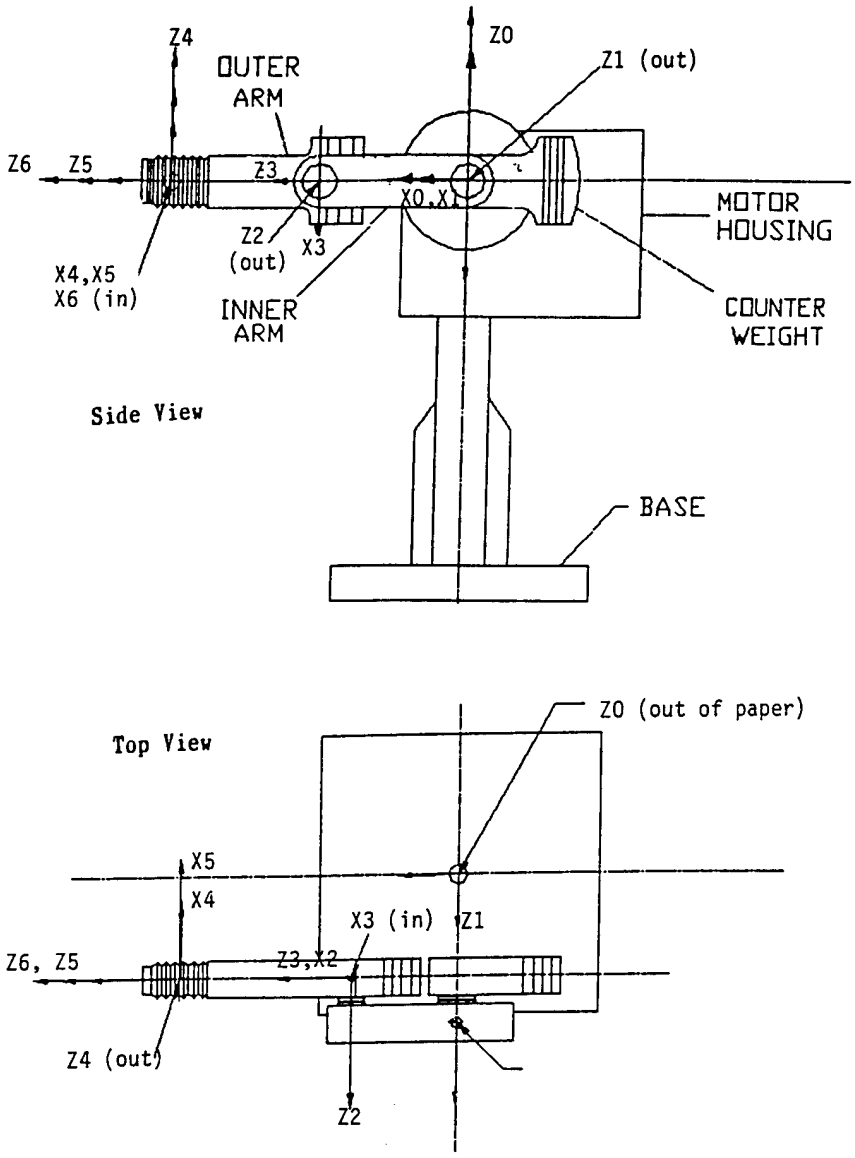
A MERLIN 6500 robot (by American Robotics Corp.) is used as the slave manipulator which is controlled by the operator through the FREFLEX to interact with the environment. The robot is a six-degree-of-freedom, revolute-jointed industrial robot with stepper motor drives and a geometry similar to that of the commonly known PUMA manipulator. Typically the MERLIN can be programmed to operate using its own specific programming system, AR-SMART [8], provided by the manufacturer. In order to achieve the necessary speed of response, the programming system is bypassed and a direct communication link is implemented between the FREFLEX and the MERLIN controllers through a high speed interface protocol [8].

During system operations, Cartesian position and orientation ("pose") data of the FREFLEX master is sent to the MERLIN controller which, in turn, must drive the slave robot to the same input position and orientation (which, of course, is now relative to the slave base frame). Obviously, for the MERLIN controller, it is necessary to compute the corresponding joint angles for a given pose command. Again, since these joint angles must be updated at every sampling time as new commands to joint servos, the computations must be made as efficient as possible.

In the following, we present for completeness the results from the inverse kinematics analysis developed for the MERLIN to facilitate efficient computations. While details of analysis in arriving at the solutions will not be reported here, we note that the approach taken follows the one referenced in earlier discussion, in which the kinematic equation is separated into translational and rotational parts. For MERLIN, we have obtained in explicit closed form the resulting set of joint angle solutions. Based on the frames and D-H parameters defined in Figure 5, the complete solutions are given as follows.

Let the desired position transformation be given by:

$$ {}^{0}\mathbf{T}_{6} = \begin{bmatrix} \mathbf{Q} & \mathbf{r} \\ \mathbf{0} & 1 \end{bmatrix} = \begin{bmatrix} q_{11} & q_{12} & q_{13} & r_1 \\ q_{21} & q_{22} & q_{23} & r_2 \\ q_{31} & q_{32} & q_{33} & r_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} $$



	a_i	d_i	α_i	θ_i
1	0	0	-90^o	θ_1
2	a_2=17.38"	d_2=12.00"	0^o	θ_2
3	0	0	$+90^o$	θ_3
4	0	d_4=17.24"	$+90^o$	θ_4
5	0	0	-90^o	θ_5
6	0	d_6	0^o	θ_6

Figure 5. Schematic of MERLIN Robot and Definition of Kinematic Frames with Its Corresponding Table of D-H Parameters

And define:

$$h = \begin{bmatrix} r_1 - q_{13}d_6 \\ r_2 - q_{23}d_6 \\ r_3 - q_{33}d_6 \end{bmatrix} = \begin{bmatrix} h_1 \\ h_2 \\ h_3 \end{bmatrix}$$

Then

$$\theta_1 = \operatorname{atan2}(h_2, h_1) - \operatorname{atan2}\left(\frac{d_2}{\eta}, \sigma_1\sqrt{1-\left(\frac{d_2}{\eta}\right)^2}\right)$$

where $\eta = \sqrt{h_1^2 + h_2^2}$; $\sigma_1 = \pm 1$

$$\theta_3 = \operatorname{atan2}(s_3, c_3) \qquad \text{where } f = h_1\cos\theta_1 + h_2\sin\theta_1 \text{ ; } \sigma_2 = \pm 1$$

$$s_3 = \frac{f^2 + h_3^2 - a_2^2 - d_4^2}{2a_2d_4} \quad ; \quad c_3 = \sigma_2\sqrt{1-s_3^2}$$

$$\theta_2 = \operatorname{atan2}(s_2, c_2) \qquad \text{where}$$

$$s_2 = \frac{fd_4c_3 - h_3(a_2 + d_4s_3)}{h_3^2 + f^2} \quad ; \quad c_2 = \frac{h_3d_4c_3 - f(a_2 + d_4s_3)}{h_3^2 + f^2}$$

Next form:

$$\mathbf{B} = \begin{bmatrix} b_{11} & b_{12} & b_{13} \\ b_{21} & b_{22} & b_{23} \\ b_{31} & b_{32} & b_{33} \end{bmatrix} = \begin{bmatrix} c_1c_{23} & s_1c_{23} & -s_{23} \\ -s_1 & c_1 & 0 \\ c_1s_{23} & s_1s_{23} & c_{23} \end{bmatrix} \mathbf{Q}$$

Then:

$$\theta_4 = \operatorname{atan2}\left(\frac{\sigma_3 b_{23}}{\sqrt{b_{13}^2 + b_{23}^2}}, \frac{\sigma_3 b_{23}}{\sqrt{b_{13}^2 + b_{23}^2}}\right) \quad ; \quad \sigma_3 = \pm 1$$

$$\theta_5 = \operatorname{atan2}\left(-b_{13}c_4 - b_{23}s_4 \, , \, b_{33}\right)$$

$$\theta_6 = \operatorname{atan2}\left(\left(b_{21}c_4 - b_{11}s_4\right), \left(b_{22}c_4 - b_{12}s_4\right)\right)$$

It is important to note that the computations should be carried out in the same order as presented above.

It might be helpful to make a few general comments concerning the practical implementation of the inverse kinematics solution. Note that, although in theory multiple solutions are possible to achieve a given end-effector position (e.g., a total of eight solutions for the MERLIN geometry), restrictions from physical and/or operational constraints will often render fewer feasible solutions in reality. As a case in point, the MERLIN used in this study was physically constrained to maintain a left-shoulder configuration, stipulating σ_1 being always positive. If, in addition, an elbow-up constraint (σ_2 negative) is also imposed, the number of feasible solutions then reduces to two, given by the remaining two values of s_3 corresponding to different configurations of the wrist assembly. Which of the two to pick can depend on operational consideration, such as selecting the one closest to the current positions, namely the one which results in the minimum joint motions.

Another seemingly trivial but nevertheless useful note concerns the way in which the joint angle command is expressed. As a rule of good practice, one should express the angles returned from trigonometric operation (e.g., atan2) in a sign consistent manner, i.e., between +/- 180 degrees, particularly when they are to be used as joint commands. This arises from the fact that most joint servos are direction-sensitive to the signs of angles; angular joint commands of, say, 150 degrees and -330 degrees are likely to cause the motor to turn in opposite directions, although they both will reach the same angular position eventually.

DISCUSSION AND CONCLUSION

We have presented the development of coordination algorithms for control implementation of a telemanipulation system consisting of an exoskeleton master and an articulated slave manipulator. Physical modeling to analyze mechanics of each subsystem was established, and algorithms for its coordination were subsequently formulated. Throughout the development, special emphasis was placed on computation efficiency due to its role in system performance.

The algorithms presented in this report have been fully tested and implemented on the robot subsystems and, as a result, significant improvements in the overall system performances have been achieved. In an initial implementation on the FREFLEX, using the same processor (68020), a greater than five-fold speed increase in computation cycle time (from 90 down to 16 msec) was achieved using the present algorithm as compared to an earlier one. One can further improve the sampling rate by (1) using a faster processor, and (2) updating the gravity torque commands (the most computationally intensive part) less frequently. The latter is justified in light of slowly changing configurations of the FREFLEX when being controlled by the operator. Both of the above steps have been done and, as of this writing, a FREFLEX motor torque update rate of 287 Hz has been attained using a newly replaced 68030 processor and updating gravity torque commands once every 10 cycles. The MERLIN inverse kinematics solution is computed on a 68030 processor in less than the minimum 4 ms MERLIN controller update period.

REFERENCES

1. T. B. Sheridan, Telerobotics, Automation and Human Supervisory Control, MIT Press, Cambridge MA, 1992.
2. A.K. Bejczy and Z. Szakaly, "Universal Computer Control System for Space Telerobotics," Proc. of IEEE Conference on Robotics and Automation, Vol. 1, Raleigh NC, 1987.
3. G. Hirzinger, J. Dietirch, and B. Brunner, "Multisensory Telerobotic Concepts for Space and Underwater Applications," Proc. of the Space and the Sea Colloquium, Paris, France, 1990.
4. R. P. Paul, Robot Manipulators, MIT Press, Cambridge MA, 1981.
5. K. S. Fu, R.C. Gonzalez, and C.S.G. Lee, Robotics: Control, Sensing, Vision, and Intelligence, McGraw Hill, 1987.
6. Odetics, Inc., "Exoskeleton Master Arm, Wrist, and End-Effector Controller with Force-Reflecting Telepresence," U.S. DoD, SBIR Program, Phase II Final Report, December 1992.
7. J.-J. Lee and L.-W. Tsai, "The Structure Synthesis of Tendon-Driven Manipulators Having a Pseudotriangular Structure Matrix," The International Journal of Robotics Research, Vol. 10, No. 3, June 1991.
8. MERLIN^{TM} System Operator Guide, Version 3.0, American Robot Corp., June 1985.

U. S. Government Printing Office 1996 750-071/00159